Paper ID: 132

15th ICCRTS

"The Evolution of C2"

Title of Paper:

Mission-Dependent Trust Management in Heterogeneous Military Mobile Ad Hoc Networks

Topic(s):

2-Networks and Networking

5-Experimentation and Analysis

6-Modeling and Simulation

Name of Author(s):

Jin-Hee Cho, Ph.D.
Army Research Laboratory – CISD

Ananthram Swami, Ph.D.
Army Research Laboratory – CISD

Ing-Ray Chen, Ph.D.
Virginia Tech – Department of Computer Science

Point of Contact:

Jin-Hee Cho, Ph.D.
Army Research Laboratory – CISD
2800 Powder Mill Rd, Adelphi, MD 20783
(Office) 301-394-0492
jinhee.cho@us.army.mil, jinheechogwb@yahoo.com

Report Documentation Page	*Form Approved* *OMB No. 0704-0188*

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **JUN 2010**	2. REPORT TYPE	3. DATES COVERED **00-00-2010 to 00-00-2010**

4. TITLE AND SUBTITLE **Mission-Dependent Trust Management in Heterogeneous Military Mobile Ad Hoc Networks**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Army Research Laboratory ? CISD,2800 Powder Mill Rd,Adelphi,MD,20783**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR’S ACRONYM(S)
	11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

Proceedings of the 15th International Command and Control Research and Technology Symposium (ICCRTS ’10), Santa Monica, CA, June 22-24, 2010

14. ABSTRACT

Managing trust in a tactical Mobile Ad Hoc Network (MANET) is challenging when collaboration or cooperation is critical to achieving military missions and system goals. Further, in heterogeneous MANETs, evaluating the trust level of a mission-driven group communication system accurately and identifying well-qualified nodes to perform a given mission are also crucial for successful mission completion. Some nodes will be highly qualified to perform the mission while others will not be, depending on the mission. Based on the context-dependent characteristic of trust, we propose a mission-dependent trust management protocol that dynamically evaluates the trust values of nodes, and dynamically recomposes the mission team so as to maximize the mission success probability. We develop a composite trust metric that considers aspects of QoS trust derived from communication networks and social trust derived from social and cognitive networks. We show that the proposed mission-dependent trust management protocol outperforms a unified trust management protocol which is not customized for performing a particular mission.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**17**	

Mission-Dependent Trust Management in Heterogeneous Military Mobile Ad Hoc Networks

Abstract

Managing trust in a tactical Mobile Ad Hoc Network (MANET) is challenging when collaboration or cooperation is critical to achieving military missions and system goals. Further, in heterogeneous MANETs, evaluating the trust level of a mission-driven group communication system accurately and identifying well-qualified nodes to perform a given mission are also crucial for successful mission completion. Some nodes will be highly qualified to perform the mission while others will not be, depending on the mission. Based on the context-dependent characteristic of trust, we propose a mission-dependent trust management protocol that dynamically evaluates the trust values of nodes, and dynamically recomposes the mission team so as to maximize the mission success probability. We develop a composite trust metric that considers aspects of QoS trust derived from communication networks and social trust derived from social and cognitive networks. We show that the proposed mission-dependent trust management protocol outperforms a unified trust management protocol which is not customized for performing a particular mission.

1. Introduction

Mobile ad hoc networks (MANETs) are defined as multi-hop wireless networks dynamically formed by mobile nodes without the help of any centralized infrastructure [9]. Unlike stationary wired networks, security protocol designers for MANETs face technical challenges due to the unique characteristics of MANETs such as resource-constraints (e.g., bandwidth, memory, energy, and, computational power), openness to eavesdropping, high security threats or vulnerabilities, inherent unreliable communications of the wireless medium, and rapid changes in topologies or memberships due to node mobility or failure [9]. Group communication systems (GCSs) in MANETs, such as in military battlefields or emergency rescues, require teamwork and collaboration to achieve a mission that depends on the trust relationships among group members [13].

The concept of "trust" originally derives from the social sciences and is defined as the degree of subjective belief about the behaviors of a particular entity [7]. Blaze *et al*. [4] first introduced the term *trust management* and identified it as a separate component of security services in networks. Since its inception, trust management in MANETs (TMM) also has received considerable attention due to its crucial necessity and diverse applicability in the decision making process. TMM is needed when participating nodes, without previous interactions, desire to establish a network with an acceptable level of trust relationships among themselves, for example, in building initial trust bootstrapping, or coalition operation without predefined trust.

A given mission may require nodes with specific properties reflecting aspects of Quality-of-Service trust (QoST) and social trust (ST). Some nodes will be highly qualified to perform the mission while others will not be, depending on the mission. Therefore, based on the context-dependent characteristic of trust, we propose a mission-dependent trust management protocol where a mission team consists of best qualified nodes that are dynamically selected upon every trust update with the goal of maximizing the mission success probability.

The contributions of this work are as follows. First, we propose a mission-dependent trust management (MDTM) suitable for tactical MANETs based on context-dependent characteristics of trust. Second, our proposed protocol dynamically evaluates the trust values of nodes, and dynamically recomposes the team so as to maximize the probability of mission success. Third, we propose a novel composite trust metric consisting of *QoS trust* derived from communication networks and *social trust* derived from social and cognitive networks. Fourth, we develop a new reliability metric, called the mission success probability, based on the trust levels required by a given mission. Fifth, we develop a novel "continuous-time" hierarchical modeling technique based on Stochastic Petri Nets (SPN) to describe and evaluate the proposed MDTM protocol; this facilitates handling of dynamics and results in a scalable implementation. Finally, we show that the proposed MDTM protocol outperforms a unified trust management (UFTM) protocol which is not customized for performing a particular mission.

The rest of the paper is organized as follows. Background and related work are discussed in Section 2. The composite trust metric, computation of the proposed mission-success probability metric, attack and energy models, and underlying assumptions are described in Section 3. The SPN-based performance model developed for analyzing performance characteristics of MDTM is the focus of Section 4. Analytical results are provided in Section 5 where the proposed MDTM is compared with the UFTM. A discussion of the limitations of this work and directions for future work are provided in Section 6.

2. Related Work

Due to the unique characteristics of MANETs and the inherent unreliability of wireless communications, trust management for MANETs should be dynamic and account for uncertainty. Trust is context-dependent, and subjective, and not necessarily transitive or reciprocal. The context-dependent characteristic of trust is popularly discussed with a typical example that Alice may trust Bob to order wine at dinner but wouldn't trust him to fix her car [1]. Similarly in MANETs depending on the given task, different types of trust (e.g., trust in computational power or trust in unselfishness) are required. The reader is referred to [5] for a detailed discussion of properties and references on trust. Trust management systems for MANETs should consider the following design features: trust metrics must be customizable; evaluation of trust should be fully distributed without reliance on a centralized authority, and should cope with dynamics and adverse behaviors in a tactical MANET [5].

Most of the existing trust management or reputation schemes focus on "context-awareness" in formulating the trust level of an entity. Context information is defined as any information that explains the condition of an entity in terms of a specific aspect [23]. Further, a context refers to the set of all context information including the characteristics of the entities that are appropriate for a particular task with their appropriate aspects [23]. Adams *et al*. [1] considered context information to evaluate the trust level in determining access rights to a decentralized system. Corradi *et al*. [8] proposed a trust model adapting trust relationships to dynamic context information occurring in pervasive computing environments. They also incorporated dynamic context to evaluate the trust level of an entity. Gray *et al*. [12] integrated trust-based admission control with standard role-based access control. Moloney and Weber [16] proposed a context-aware trust-based security system for MANETs. Tavakolifard *et al.* [23] proposed a context-aware trust model based on the fact that behaviors of an entity are influenced by the situation. Uddin *et al.* [25] proposed an interaction-based context-aware trust model for open and dynamic systems where services are regarded as contexts. Our work differs from the above in that we select a set of qualified nodes to meet context-dependent mission requirements.

Bertocco and Ferrari [2] proposed a reputation system for a centralized system where an agent can join a group with particular context information in which it is interested. Similarly, Billhardt *et al*. [3] examined how to select appropriate service providers based on trust when there is no prior experience with the service providers. Toivonen *et al.* [24] investigated trust levels where trustors are placed in diverse situations. They claimed that trust can vary depending on the situation of the trustor and proposed functions to determine context-aware trust. Our work differs from the above work in that context is used as a mission requirement to select best qualified nodes to maximize the mission success probability. Further, we consider two different aspects of trust, namely, QoS trust and social trust.

Namuduri [17] proposed an active trust model for an airborne network requiring trust assessment before sharing mission-specific information during a short mission execution period based on zero-knowledge proofs. This work is similar to our work in that trustable nodes are selected for executing a mission by sharing mission specific-information. However, Namaduri's model assesses trust based solely on whether or not a node possesses a secret key, whereas we use a composite trust metric consisting of QoS trust and social trust.

The research area of resource allocation, e.g., matching sensors with missions, uses ideas similar to ours although they do not deal with trust. Mainland *et al.* [15] proposed market-oriented methods to optimize system performance in terms of resource allocation. Preece *et al.* [19] investigated how to match available resources, particularly sensors, in battlefield situations to achieve efficient mission completion. Preece *et al.* [20] also investigated sensor-mission assignments in the context of dynamic coalitions and changing mission requirements. Rowaihy *et al.* [21] proposed centralized and distributed schemes to assign sensors to dynamically changing environments. Wang *et al.* [26] dealt with context-aware service matchmaking using description logic. However, the body of work described above does not deal with trust.

Recently, we proposed a trust management protocol for a cognitive GCS in MANETs; our trust metric was composed from QoS trust and social trust [6]. However, the prior work assumes a static team; it does not deal with dynamic selection of team members as the trust status of the network evolves.

3. System Model

In the initial network deployment, we assume that there is no predefined trust. Without prior interactions, the initial bootstrapping will establish a shallow level of trust based only on limited direct observations, indirect information through third parties, and authentication by a challenge/response process. Over time, participating nodes will establish a stronger trust level with more confidence based on direct or indirect interactions. Our trust management protocol allows each node to evaluate the trust levels of other nodes as well as to be evaluated by other nodes.

Trust decays over time without further updates or interactions between entities. Node mobility also may hinder continuous interactions with other group members, lowering the chances of evaluations of each other in the group. This includes cases such as a node moving to other areas causing its disconnection from the current group, leaving a group for mission reasons, voluntary disconnection for saving power or involuntary disconnection due to physical terrain or low energy. On the other hand, node mobility could enhance trust evaluation of distant nodes. We use the concept of a trust chain of a node to indicate derivation of indirect trust evidence across multiple-hops; however, one should expect that the degree of trust would decay as the length of the trust chain increases.

Our target system is a mission-driven GCS in military tactical MANETs where a symmetric key, called the group key, is used as a secret key for group communications between group members [14]. Upon a node's disconnection from the group, the system generates and redistributes a new key so that non-member nodes will not be able to access a valid secret group key. Nevertheless, each group member keeps old trust information even for non-member nodes so that the information can be reused for future interactions. This is useful to cope with potential newcomer attackers who flush their low trust levels by frequently rejoining the group. Our trust metric will span two aspects of the trust relationship. First, *social trust* [11] will be evaluated through social networks. Social networks may be neighboring nodes or any nodes that had previous relationships due to their common interests. Based on the generally accepted definition of social networks [11], they may not necessarily be the same as networks that perform the task for mission execution. Second, *quality-of-service (QoS) trust* evaluated through information networks accounts for the capability to complete the assigned mission.

The key question of this work is "Can we trust this node to do mission X?" To effectively answer this question, we develop mission-dependent trust management (MDTM) for a tactical GCS in MANETs. The underlying idea is that we select the best qualified nodes to perform a given mission which requires certain characteristics. For example, when a mission requires high degree of the properties related to QoS trust (e.g., energy level or cooperation or timeliness), we use criteria with high priority to the QoS aspects of trust. On the other hand, if the mission rather requires high degree of social trust properties (e.g., honesty, proximity, betweenness), we use criteria emphasizing the social aspects of trust.

3.1 Assumptions and Design

We assume a pure MANET environment, without a centralized trusted entity, where nodes communicate through multi-hops. Members of a GCS in MANETs are heterogeneous nodes consisting of typical types of military nodes such as robots equipped with sensors, dismounted soldiers carrying mobile devices, unmanned vehicles with mobile devices, and manned vehicles with mobile devices [18]. The heterogeneous group member nodes have different capabilities or characteristics in terms of platform functionalities, moving speed, energy level, and intrinsic tendencies of selfish or malicious behaviors. In this work, we considere 4 types of nodes where higher type nodes (e.g., node type 4 represented as NT4) have higher energy level and speed than lower type nodes (e.g., node type 1 represented as NT1). The initial energy level and speed are assumed to be uniformly distributed and mutually independent. However, we do not link intrinsic tendencies of selfish or malicious behaviors of nodes with the node type. The intrinsic tendencies of selfish and malicious behaviors are modeled as exponentially distributed random variables, representing the probability that a node will behave selfishly or not and maliciously or not. We also associate the energy level of a node with the intrinsic tendencies of selfish or malicious behaviors at runtime. That is, even if a node has intrinsically very bad behavioral characteristics, these behaviors can be relaxed and the node can become generous when it is under less stressful environments such as having a sufficient amount of energy. Thus, depending on the given intrinsic tendencies, the degree of being good (e.g., cooperative or honest) or bad (e.g., selfish or dishonest by being compromised) at runtime can also vary.

Even if the intrinsic tendencies of selfishness and maliciousness may affect the behaviors of nodes, the environmental conditions such as remaining energy will also affect the behaviors of the nodes. A node is more likely to be selfish when it has low energy and vice versa. Further, a node is more likely to be compromised when it has low energy and vice versa, since a node with high energy is more capable of defending itself against attackers by using more energy-consuming defense mechanisms.

Each node periodically beacons heartbeats with its *id* and *location* information so that node failure or disconnection is easily detected and accordingly immediate rekeying operation can be performed to maintain secrecy properties (i.e., backward or forward secrecy). We assume that there exists a distributed intrusion detection subsystem (IDS) for detecting insider attacks. As soon as a node is detected by IDS, we assume that the node is no longer available in the system, meaning that trust value of the node will drop suddenly. We do not make any assumptions about the IDS, except that it has known false positive and false negative probabilities. We define the selfish behavior of a node as dropping group communication packets transmitted from other nodes. Thus, even though the node is selfish, it cooperates to perform rekeying and IDS-related operations. We also assume that potential attackers, compromised but not detected by IDS, may disseminate bogus packets to perform attacks such as fake information dissemination.

The energy level of each node is adjusted depending on its status. For simplicity, we only consider energy consumption based on communication type, receiving or transmitting. For example, if a node becomes selfish, the rate of energy consumption is slowed down. If a node becomes compromised but not detected by IDS, the rate of energy consumption could grow since the node may have a chance to perform attacks, thus consuming more energy. We only consider redemption mechanism for selfish nodes. At the end of a reevaluation period, which corresponds to a trust update interval, selfish nodes will decide whether they will resume normal behaviors or continue being selfish depending on their own energy level. In addition, when a node is not a member, it will not consume as much energy as when it is a member. We model group member join and leave operations as most GCSs have. Upon every membership change due to join/leave/eviction, individual rekeying will be performed based on a distributed key agreement protocol.

We assume that a node's trust value is assessed based on direct observations as well as indirect observations. For indirect observations, we use recommendations obtained from 1-hop neighbors with the *k* highest trust values on the trust chain. If *k* recommenders are not found, recommendations from all 1-hop neighbors can be used. The trust value is updated by exchanging status information periodically. The status exchange packet includes a node's own information as well as information of nodes on its trust chain.

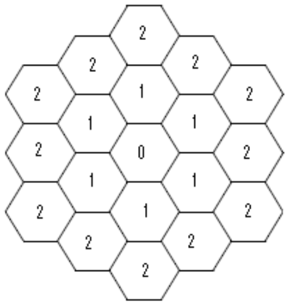

Figure 1: Hexagonal Network Model.

We adopt the hexagon-shaped operational area consisting of $3m^2 + 3m + 1$ sub-hexagon areas where m indicates a ring level = 0, 1, 2, etc. Figure 1 shows an example with *m* = 2. The diameter of a hexagonal area equals the wireless radio range (*R*). In order to analytically model average behaviors of nodes, we assume that a node is located in the center of a hexagon area. A node in area 0 (called a ring level 0) can communicate with nodes in area 1 (called a ring level 1) with 1-hop distance. In this case, we regard the nodes in areas 1 and 0 (its current location) as 1-hop neighbors of the node located in area 0. The *n*-hop neighbors are calculated similarly. Each node moves toward a target area designated by a given mission. We model this by allowing a node to randomly select areas closer to the target area with high probability and the areas distant from the target area with lower probability.

Pham *et al.* [18] describe typical mission teams in military battlefield MANETs. Similarly, we also assume that a mission team is deployed with team members who are best qualified for a given mission execution during a short period of time. In particular, upon every trust update, a different set of members will be selected based on the characteristics required by the mission. We select the best *k* qualified nodes with the top highest trust values from the nodes in each node type with $k < N/N_{NT}$, where *N* is the total number of members, and N_{NT} is the number of nodes types. The criteria to select nodes for mission execution will be either MDTM or UFTM. The goal of this study is to compare the proposed MDTM with UFTM in terms of the mission success probability.

3.2 Composite Trust Metric

We consider a trust metric that spans two aspects of the trust relationship: QoS trust and social trust. The overall trust value is calculated based on the degrees of energy and cooperation for QoS trust and the degrees of honesty, proximity to a mission-designated target area, and betweenness centrality for social trust, with a fixed ratio (QoS trust : social trust). The values of each trust component and the overall trust value will be in the range of [0, 1], with 0 indicating complete distrust, 0.5 ignorance and 1 complete trust. A node's trust value changes dynamically to account for trust decay over time due to node mobility or failure, as the trust chain becomes longer, as the node's energy level changes, and as the node becomes compromised or selfish.

Based on the trust values calculated by 1-hop neighbors, the trust value can be calculated by *n*-hop neighbors. The information used for trust evaluation of a particular node *j*, the trustee, by a particular node *i,* the trustor, includes probability of being alive if remaining energy > energy threshold ($T_{i,j}^{n-hop,energy}(t)$), probability of being cooperative ($T_{i,j}^{n-hop,cooperation}(t)$), probability of being honest ($T_{i,j}^{n-hop,honesty}(t)$), relative average closeness from node *j*'s location to a target area ($T_{i,j}^{n-hop,proximity}(t)$), and relative average closeness from node *j*'s location to all other nodes ($T_{i,j}^{n-hop,betweenness}(t)$) , where *n* is the length of the trust chain (TC) of a node and *t* is time. The basis (i.e., direct observation to derive 1-hop local trust value) for the computation of these five trust component values will be explained in Section 4. We refer the reader to our prior work [6] for details of the technical method for obtaining them from a SPN performance model.

Below we detail how the trust value is calculated. The *n*-hop trust value of node *j* by node *i*, $T_{i,j}^{n-hop}(t)$, is calculated using direct observations and indirect information forwarded from recommenders who are 1-hop neighbors of node *i,* using a trust chain of length *n*. If the length of the trust chain is *n*, then in evaluating the trust value of node *j*, node *i* considers recommendations from k_{recom} 1-hop neighbors incorporating information passed from all nodes on its trust chain that are within *n* hops of itself. Note that a node, say *k*, in this *n*-hop neighborhood of node *i*, will also compute its trust value of node *j*, based on direct and indirect evidence collected by nodes in node *k*'s *n*-hop neighborhood. Specifically, $T_{i,j}^{n-hop}(t)$ is defined by:

$$T_{i,j}^{n-hop}(t) = P_{i,j}^{n-hop}(t)\left[\begin{array}{c} \beta_1\left(\dfrac{T_{i,j}^{n-hop,energy}(t) + T_{i,j}^{n-hop,cooperation}(t)}{2}\right) + \\ (1-\beta_1)\left(\dfrac{T_{i,j}^{n-hop,proximity}(t) + T_{i,j}^{n-hop,honesty}(t) + T_{i,j}^{n-hop,betweenness}(t)}{3}\right)\end{array}\right] \quad (1)$$

Here β_1 and $(1-\beta_1)$ are the non-negative weights for QoS trust and social trust respectively. In Equation 1, all the $T_{i,j}$ terms will be in the range of [0, 1] and each trust component in QoS trust and social trust will be equally weighted. For the proposed MDTM, we will vary β_1 to reflect the priority of the required characteristics, either QoS trust or social trust. The *n*-hop trust values are computed from the *(n-1)*-hop and indirect trust values via:

$$T_{i,j}^{n-hop,Z}(t) = \alpha T_{i,j}^{(n-1)-hop,Z}(t) + (1-\alpha)T_{i,j}^{n-hop,Z-indirect}(t) \quad (2)$$

Equation 2 explains how each *n*-hop trust component (where Z is energy, cooperation, honesty, proximity, or betweeness) is calculated from its *(n-1)*-hop "self-information" with weight α for the node's own information, and weight $(1-\alpha)$ for indirect information, say "other-information," in the *n*th hop. $T_{i,j}^{(n-1)-hop,Z}(t)$ can be obtained by recursively using Equation 2. $T_{i,j}^{n-hop,Z-indirect}(t)$ is calculated as:

$$T_{i,j}^{n-hop,Z-indirect}(t) = \sum_{k\in K}\left[\left(\frac{T_{i,k}^{1-hop,Z}(t)}{\sum_{k\in K} T_{i,k}^{(n-1)-hop,Z}(t)}\right)T_{k,j}^{(n-1)-hop,Z}(t)\right] \quad (3)$$

In Equation 3, *K* is the set of the *id*s of recommender nodes that have the highest trust values among all 1-hop neighbors on the trust chain where $|K| = k_{recom}$. Notice that in calculating the trust value of node *j* by node *i* via node *k*'s recommendation, node *i*'s trust value on node *k* (with a denominator that sums up *i*'s trust values on all recommenders) is used as a weight.

In Equation 1, $P_{i,j}^{n-hop}(t)$ is the probability that nodes *i* and *j* are within *n* hops and is used to take into account the amount of accumulated interactions between nodes *i* and *j* within *n* hops. . That is, node *i* considers its experiences with node *j* when they are within *n* hops of one another. The intuition is that longer and more interactions will increase the certainty of the trust relationship. $P_{i,j}^{n-hop}(t)$ is computed by:

$$P_{i,j}^{n-hop}(t) = \frac{\sum_{h=1}^{n} q_{i,j}^{h-hop}(t)}{\sum_{h=1}^{max} q_{i,j}^{h-hop}(t)} \tag{4}$$

$$where\ q_{i,j}^{h-hop}(t) = \sum\nolimits_{(l,m)\epsilon S}\left(P_i^{loc=l}(t)\ P_j^{loc=m}(t)\right) \text{ and } n \leq max$$

Here S is a set covering all (l, m) pairs with the distance between l and m being k-hops, and max is the maximum length of the trust chain of a node. Notice that $P_{i,j}^{n-hop}(t)$ is the probability that the hop distance between two nodes is less than or equal to n while $q_{i,j}^{k-hop}(t)$ is the probability that the hop distance between the two nodes is *exactly k*. $P_i^{loc=k}(t)$ refers to the probability that node i is located in area k. By dividing by ($\sum_{h=1}^{max} q_{i,j}^{h-hop}(t)$), we consider the relative largeness of the interactions that have occurred.

Observe that the n-hop trust values are calculated based on *(n-1)*-hop trust values. Thus, the basis of all trust values with the trust chain length n > 1 is the 1-hop trust value, hereafter called the *1-hop local trust value*. The 1-hop local trust value of node j evaluated by node i for a particular trust component Z, $T_{i,j}^{1-hop,Z}(t)$, is computed by:

$$T_{i,j}^{1-hop,Z}(t) = min\left[\frac{T_j^Z(t)}{T_i^Z(t)}, 1\right] \tag{5}$$

Equation 5 considers the subjective characteristic of trust by evaluating a trustee (node j) based on a trustor (node i)'s own standard. The trustor gives a perfect score of 1 when the trustee's trust value is larger than its own trust value. Otherwise, node i scales it down to its own standard. $T_j^Z(t)$, where Z is energy, cooperation, or honesty, can be obtained from our SPN model shown in Section 4.2; for technical details, we refer the reader to our prior work [6]. For Z = betweenness or proximity, Equations 7 and 8 below show how to compute $T_j^Z(t)$ based on location information.

Notice that in Equation 3, $T_{i,j}^{n-hop,Z-indirect}(t)$ is calculated based on the *(n-1)*-hop trust values such as $T_{i,k}^{(n-1)-hop,Z}(t)$ and $T_{k,j}^{(n-1)-hop,Z}(t)$. However, for $T_{i,j}^{1-hop,Z-indirect}(t)$, the local trust value with n = 1 does not have *(n-1)*-hop trust values, and thus we calculate it based on the 1-hop trust values at time $(t-\Delta)$. This is a reasonable choice to reflect past experiences for trust evaluation. Thus, $T_{i,j}^{1-hop,Z-indirect}(t)$ is given by:

$$T_{i,j}^{1-hop,Z-indirect}(t) = \sum_{k\in K}\left[\left(\frac{T_{i,k}^{1-hop,Z}(t-\Delta)}{\sum_{k\in K} T_{i,k}^{1-hop,Z}(t-\Delta)}\right) T_{k,j}^{1-hop,Z}(t-\Delta)\right] \tag{6}$$

Except for using the past experience at time $(t-\Delta)$ with 1-hop trust values, other notations and physical meanings are the same as for Equation 3.

Based on $P_j^{loc=k}(t)$ (the probability that node j is located in area k) obtained as above, the probability that two nodes are k-hop away can be computed. $T_j^{proximity}(t)$ and $T_j^{betweenness}(t)$ can then be computed based on location information as follows:

$$T_j^{proximity}(t) = \sum_{i\epsilon L}\left(P_j^{loc=i}(t)\frac{(D_{max}^{target} - D(i, L_{target}))}{D_{max}^{target}}\right) \tag{7}$$

where L is a set of possible locations, D_{max}^{target} is the maximum distance to a designated target area (L_{target}) among all possible locations in the operational area and $D(i, L_{target})$ is the distance from area i to a designated target area (L_{target}) given that node j is located in area i based on Figure 1.

$$T_j^{betweenness}(t) = \frac{\sum_{i\epsilon L}\sum_{h\epsilon M}\sum_{k\epsilon L}\left(P_j^{loc=i}(t)P_h^{loc=k}(t)\frac{(D_{max} - D(i,k))}{D_{max}}\right)}{|M|} \tag{8}$$

where *M* is a set of all nodes' *id*s, $|M|$ is the number of nodes in set *M*, D_{max} is the maximum distance among the distances between node *j*'s location and all possible locations of node h and $D(i,k)$ is the distance from node *j*'s location to node *h*'s location given that node *j* is located in area *i* and node *h* is located in area *k* based on Figure 1.

3.3 Mission Success Probability

We define the mission success probability as a reliability metric based on the trust level required for successful mission execution. First of all, we calculate the reliability *R(t)* in the same way that the reliability of a serial system with multiple components is calculated [22], as the product of the component reliabilities. Here each component is the reliability of team members belonging to the same node type. Thus, *R(t)* is computed as:

$$R(t) = \prod_{v=1}^{m} R_{NT_v}^{k-out-of-n}(t) \ \ where\ k = ceil(\frac{2}{3} * n) \tag{9}$$

For the reliability calculation of each component, $R_{NT_v}^{k-out-of-n}(t)$, we treat it as a *k-out-of-n system* [22], meaning that the system is assumed to function properly if at least *k* subcomponents out of *n* subcomponents are operating properly. In Equation 9, *v* indicates a certain node type, *m* is the number of node types, *k* is the minimum number of properly functioning nodes in node type *v*, and *n* is the total number of nodes in node type *v*. Further, in order to derive *k* reasonably, we use the concept of Byzantine Failure (BF) [10] meaning that if more than 1/3 of the nodes are compromised, the system fails. Similarly, we define system failure if more than 1/3 of nodes do not meet the required trust thresholds: the mission fails. We apply this system failure definition in calculating the reliability of a mission team with the same node type, consisting of a set of selected mission team members. Summarizing the above, $R_{NT_v}^{k-out-of-n}(t)$ is calculated by:

$$R_{NT_v}^{k-out-of-n}(t) = \sum_{i=k}^{n} \binom{n}{k} \left(\overline{r_{NT_v}(t)}\right)^{k} \left(1 - \overline{r_{NT_v}(t)}\right)^{n-k} \tag{10}$$

$$\overline{r_{NT_v}(t)} = \frac{\sum_{j \in G} r_{NT_v}^{j}(t)}{|G|} \tag{11}$$

Here $\overline{r_{NT_v}(t)}$ is the average reliability of nodes in *G*, *G* is a set of selected nodes in node type *v* for mission execution, and $|G|$ indicates the number of selected nodes in node type *v*. For simplicity, we use the average reliability of nodes in the same node type, instead of using different node reliabilities, as shown in Equation 11.

In Equation 12, the reliability of each selected node belonging to node type *v* for mission execution ($r_{NT_v}^{j}(t)$) is calculated similar to the way we calculate trust values in Equation 1. Specifically, $r_{NT_v}^{j}(t)$ is obtained by:

$$r_{NT_v}^{j}(t) = \beta_2 \left(\frac{r_{NT_v}^{j-energy}(t) + r_{NT_v}^{j-cooperatio\ n}(t)}{2}\right) + \\ (1-\beta_2)\left(\frac{r_{NT_v}^{j-proximity}(t) + r_{NT_v}^{j-honesty}(t) + r_{NT_v}^{j-betweenness}(t)}{3}\right) \tag{12}$$

Here β_2 and $(1-\beta_2)$ are the weight parameters for QoS trust and social trust. Note that β_1 and $(1-\beta_1)$ in Equation 1 are used to indicate weights for QoS trust and social trust in calculating trust values in either MDTM or UFTM. On the other hand, β_2 and $(1-\beta_2)$ are the weights required to predict the mission success probability.

The reliability of each trust component (i.e., $r_{NT_v}^{j-Z}(t)$ where Z indicates a trust component) is calculated via the required trust thresholds, and the average trust values of a trustee node *j* by:

$$r_{NT_v}^{j-Z}(t) = \begin{cases} 1 \; if \; T_{NT_v}^{j-Z}(t) \geq D_{NT_v}^{j-Z-1} \\ 0 \; if \; T_{NT_v}^{j-Z}(t) < D_{NT_v}^{j-Z-2} \\ T_{NT_v}^{j-Z}(t)/D_{NT_v}^{j-Z-1} \; if \; D_{NT_v}^{j-Z-2} \leq T_{NT_v}^{j-Z}(t) < D_{NT_v}^{j-Z-1} \end{cases} \tag{13}$$

We use two different trust thresholds called $D_{NT_v}^{j-Z-1}$ and $D_{NT_v}^{j-Z-2}$. $D_{NT_v}^{j-Z-1}$ is the required trust level for trust component Z in node type *v*. $D_{NT_v}^{j-Z-2}$ is the system drop dead trust level for trust component Z in node type *v*. That is, if the average trust value of node *j* ($T_{NT_v}^{j-Z}(t)$) is equal to or larger than the required trust level ($D_{NT_v}^{j-Z-1}$), the reliability of node *j* ($r_{NT_v}^{j-Z}(t)$) is 1. If the average trust value of node *j* ($T_{NT_v}^{j-Z}(t)$) is less than the system drop dead trust level ($D_{NT_v}^{j-Z-2}$), the reliability of node *j* ($r_{NT_v}^{j-Z}(t)$) is 0. Otherwise, $T_{NT_v}^{j-Z}(t)$ will be scaled down based on $D_{NT_v}^{j-Z-1}$, as shown in Equation 13. Note that depending on a node type of a node, a given mission may require a different trust threshold for both $D_{NT_v}^{j-Z-1}$ and $D_{NT_v}^{j-Z-2}$. Equation 13 uses the average trust value of node *j* ($T_{NT_v}^{j-Z}(t)$) calculated by:

$$T_{NT_v}^{j-Z}(t) = \frac{\sum_{i \in M} T_{NT_v}^{i,j-Z}(t)}{|M|} \tag{14}$$

Here *M* is a set of all nodes in the network (non-selected nodes plus selected nodes for mission execution) and $|M|$ indicates the total number of elements in *M*. Note that the average trust value of node *j* in node type *v* is evaluated by all nodes in the network, not necessarily only by selected nodes in node type *v* for mission execution.

3.4 Energy Model

We follow the energy model in [6]. We associate the energy level of a node with its state: selfish/unselfish or compromised/uncompromised or whether it is a group member/non-group member. Depending on the remaining energy, each node acts differently. The degree of energy consumption is also affected by the node's state. These parameters are interwoven and affect a node's lifetime significantly. Each node must handle events such as beaconing, group communication, rekeying, and status exchange for trust update. In particular, after a status exchange event, trust evaluation of 1-hop neighboring nodes as well as distant nodes may be performed. Each node may transmit its own status (e.g., information providing the trust values) as well as trust values of other nodes on its trust chain. Recall that we use recommendations from 1-hop neighbors for trust evaluation and each status message is disseminated periodically. Further, we also distinguish inactive nodes that are not performing a mission from active nodes that are participating in a mission, and their effects are also taken into consideration to calculate energy consumption.

3.5 Attack Model

We consider the presence of outside and inside attackers by non-group members and legitimate group members. We assume that prevention techniques such as encryption, authentication, or rekeying inhibit outsider attacks. Our trust management protocol will utilize the IDS to detect inside attackers and identify compromised nodes. The IDS system will categorize nodes performing real attacks as "blacklisted" culprits and forward them to the GCS for permanent evictions of proven attackers through individual rekeying.

4. Performance Model

We develop SPN models to analyze the performance of MDTM. We use SPN because of its efficient representations of a large number of states when the underlying models are Markov or semi-Markov models. We develop a hierarchical modeling technique based on SPN to avoid state explosion problems and to improve solution efficiency for modeling a large-scale GCS operating under MDTM or UFTM.

4.1 Hierarchical Modeling using Stochastic Petri Nets

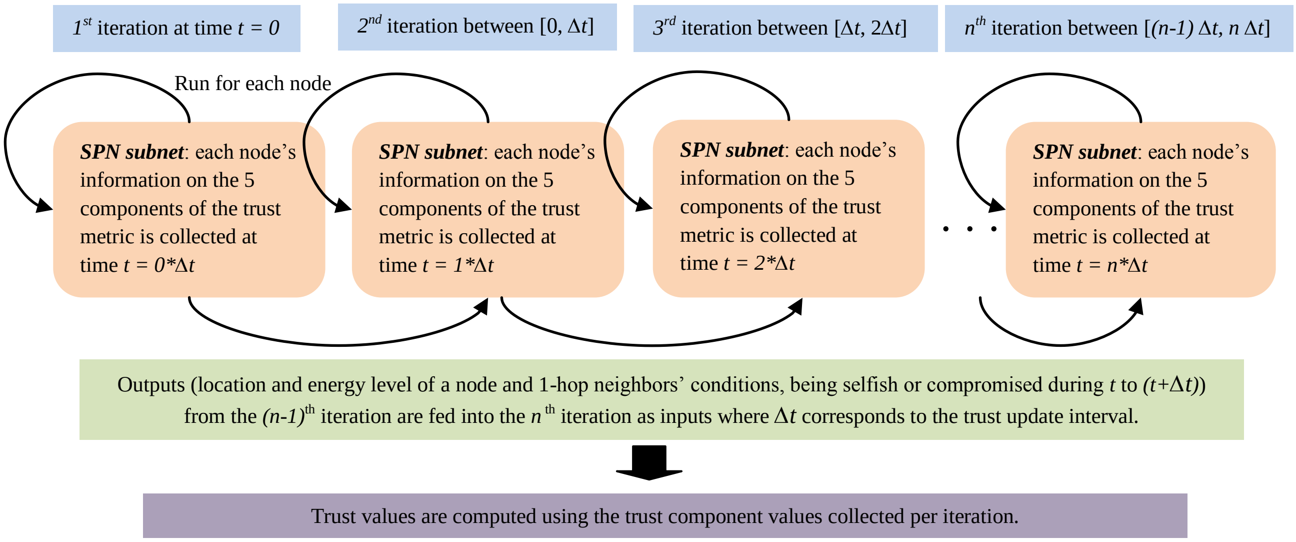
1st iteration at time $t = 0$
2nd iteration between $[0, \Delta t]$
3rd iteration between $[\Delta t, 2\Delta t]$
nth iteration between $[(n-1)\Delta t, n\Delta t]$
Run for each node
SPN subnet: each node's information on the 5 components of the trust metric is collected at time $t = 0*\Delta t$
SPN subnet: each node's information on the 5 components of the trust metric is collected at time $t = 1*\Delta t$
SPN subnet: each node's information on the 5 components of the trust metric is collected at time $t = 2*\Delta t$
SPN subnet: each node's information on the 5 components of the trust metric is collected at time $t = n*\Delta t$
Outputs (location and energy level of a node and 1-hop neighbors' conditions, being selfish or compromised during t to $(t+\Delta t)$) from the $(n-1)^{th}$ iteration are fed into the n^{th} iteration as inputs where Δt corresponds to the trust update interval.
Trust values are computed using the trust component values collected per iteration.

Figure 2: Hierarchical Modeling Processes using SPN Subnets.

We use a SPN subnet model to describe each node's lifetime; the SPN subnets do not communicate with each other directly. We use a "continuous-time'' iterative technique to capture the dynamics during an evaluation period (or an iteration). At the end of the evaluation period, each subnet yields information about the state of the node (residual energy, selfishness, location, etc.). Nodes can also obtain state information from their 1-hop neighbors at the end of the evaluation period. These pieces of information are inputs to the SPN subnet at the next evaluation period (or the next iteration).

Initially the location of each node is randomly distributed over the operational area based on the uniform distribution. As explained in our network model in Figure 1, a node moves to one of seven locations including its own location with a higher probability of moving closer to a designated target area (i.e., to the two closest with probability 0.2 each, the next two closest with 0.15, the next two closest with 0.1 and its own location with 0.1). In cases that a node's next full seven directions are not available, then equal chances are given to the available locations. The speed of each node is initially chosen randomly, depending on its node type, and is then fixed during its lifetime. Table 1 in Section 5 shows the default parameter values used. The SPN subnet for node i computes the probability that node i is in a particular hexagon area j at time t. This information along with the information of other nodes' location information at time t provides the information about a node's n-hop neighbors at time t, which we will use to compute the trust metric (see Section 3.2). Since movements are assumed to be independent, the probability that two nodes are in a particular location at time t is given by the product of the two individual probabilities. This process is done by running the SPN subnet N times for the N nodes in the network based on the continuous-time iterative technique described above.

In the first iteration, since there is no information available about 1-hop neighbors, it is assumed that each area has an equal number of nodes and all nodes are unselfish. In the second iteration, based on the information collected (e.g., location, energy level, number of cooperative/selfish and compromised/healthy 1-hop neighbors) from the first iteration, each node knows how many nodes are 1-hop neighbors that can directly communicate with it, and whether or not they are members of the GCS, cooperative or selfish, compromised (dishonest) or healthy (honest). A node also knows how many n-hop neighbors it has at time t.

4.2 SPN Models

Figure 3 shows the *SPN subnet* model for describing a node's mobility behavior, whether the node is a member or not, and a node's status in terms of its energy level, membership, degree of healthiness or honesty (e.g., whether or not a node is compromised or/and detected by IDS), and degree of unselfishness or cooperation. The SPN subnet gives the probability of each node being located in a particular area at a particular time point.

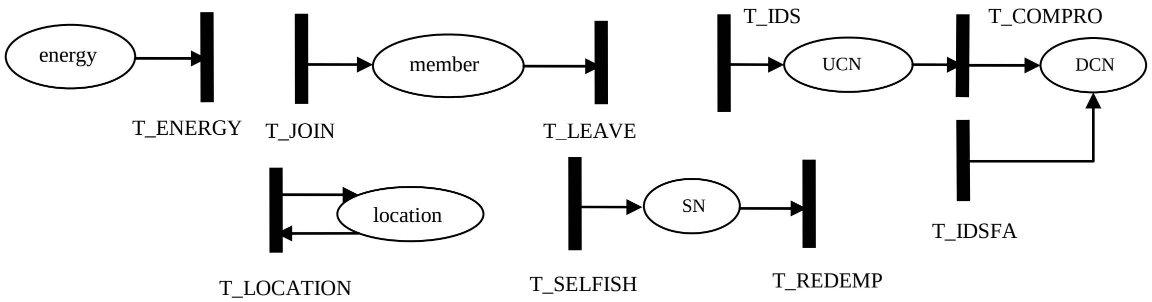

Figure 3: SPN Subnet Model.

The transition T_LOCATION is triggered when a node moves to a randomly selected area out of seven different directions including its current location, with the rate calculated as $S_{init}^{NT_v}/R$ based on an initial speed ($S_{init}^{NT_v}$) and a wireless radio range (*R*). Depending on the randomly selected location, the number of tokens in place *location* is adjusted. We assume that inter-arrival times of a node's join and leave requests are exponentially distributed with rates λ and μ respectively.

Place *energy* represents the current energy level of a node. An initial energy level is assigned according to node heterogeneity information depending on the node type. In our analytical model, we randomly generate a number between certain ranges (See Table 1 in Section 5 for details) based on the uniform distribution. A token is taken out when transition T_ENERGY fires. The transition rate of T_ENERGY is adjusted on the fly based on a node's state; it is lower when a node becomes selfish to save energy or when a node changes from member to non-member; it is higher when the node becomes compromised so that it performs attacks and consumes more energy. We assume that *T* seconds will be taken to consume one energy token when a member node has no selfish or compromised 1-hop neighbors. We follow the energy model in [6] for adjusting the time taken to consume one token in place *energy* based on a node's status.

Place *UCN* indicates an undetected compromised node. Place *DCN* represents a detected compromised node. A node is compromised when transition T_COMPRO with rate λ_{com} fires where λ_{com} is the base compromising rate initially given indicates the level of current energy. In practice, λ_{com} can be derived via first-order approximation from the observations of historical attack behaviors. The behavior of a node being compromised is associated with the energy level of the node. If the node has low energy, it is more likely to become compromised, and vice-versa. This is modeled by the enabling function of T_COMPRO, which returns 1 to enable T_COMPRO or returns 0 to disable T_COMPRO, as follows:

$$\text{enabling\_T\_COMPRO:} \qquad (15)$$

$$if(mark(energy) > 0 \ \&\& \ mark(UCN) == 0 \ \&\& \ mark(DCN) == 0 \ \&\& \ mark(member) > 0)$$

$$\{\, if(N_{rand} \ \leq P_{dis\,honest}\,)\ return\ 1;\ \ else\ return\ 0;\}$$

$$\text{where } N_{rand} \ = rand[0,1] * (mark(energy) + 1)/C_{com}$$

Here rand [0, 1] returns a random variable, $mark(energy)$ indicates the remaining energy, and C_{com} is a constant. $P_{dis\,honest}$ models the inherent behavioral nature of a node's honesty (or dishonesty) and is a randomly selected number based on the truncated exponential distribution with mean 0.5 in the range of [0, 1]. Equation 15 implies that a node behaves dishonestly based on the random seed of the bad behavior but the bad behaviors can be relaxed or further enhanced based on the current remaining energy level of the node. If the node is compromised, a token goes to *UCN*, being compromised but not detected by IDS. While the node is not detected by IDS, it has a chance to perform attacks. But right after being detected by IDS, a token is taken out from *UCN* into *DCN* and the node is evicted immediately through individual rekeying operations. We consider false alarm probabilities of IDS. False negative probability (P_{fn}^{IDS}) of IDS is applied in T_IDS with the rate $\left(1 - P_{fn}^{IDS}\right)/T_{IDS}$ where T_{IDS} is an interval triggering IDS periodically and false positive probability (P_{fp}^{IDS}) of IDS is considered in T_IDSFA with the rate P_{fp}^{IDS}/T_{IDS}.

Place *SN* represents whether a node is selfish or not. If a node becomes selfish, a token goes to *SN* by triggering T_SELFISH. Transition T_SELFISH fires based on the condition of the energy level of a node. Our assumption is that if the node has low energy, it is more likely to become selfish, and vice versa. The enabling functions for T_SELFISH and T_REDEMP are given in Equations 16 and 17 respectively by:

$$\text{enabling\_T\_SELFISH: } if(mark(energy) > 0 \ \&\& \ mark(member) > 0 \ \&\& \ mark(SN) == 0) \tag{16}$$
$$\{\ if(N_{rand} \leq P_{selfish})\ return\ 1;\ \ else\ return\ 0;\}$$
$$where\ N_{rand} = rand[0,1] * (mark(energy) + 1)/C_{selfish}$$

$$\text{enabling\_T\_REDEMP: } if(mark(energy) > 0 \ \&\& \ mark(member) > 0 \ \&\& \ mark(SN) > 0) \tag{17}$$
$$\{\ if(N_{rand} \leq P_{selfish})\ return\ 0;\ \ else\ return\ 1;\}$$
$$where\ N_{rand} = rand[0,1] * (mark(energy) + 1)/C_{selfish}$$

$P_{selfish}$ models the inherent behavioral nature of a node's selfishness and is a randomly selected number based on the truncated exponential distribution with mean 0.5 in the range of [0, 1]. Other parameters are similar to those used in Equation 15. We define T_{gc} as the time interval to disseminate a group communication packet, assumed to be exponentially distributed in this work. Each node's selfishness is checked whenever a group communication packet is transmitted, so that the transition rate of T_SELFISH is $1/T_{gc}$. The transition T_SELFISH is triggered when a node is a member, alive with remaining energy ($mark(energy) > 0$), and currently not selfish. When the randomly selected number reflecting the degree of the node's current energy level (N_{rand}) is less than the probability of selfish nature ($P_{selfish}$), T_SELFISH fires, and vice versa. We also similarly model the redemption mechanism for selfish nodes by using the transition T_REDEMP with the rate of $1/T_{trust}^{update}$. A node can have a chance to be redeemed at the end of a reevaluation period, corresponding to a trust update interval (T_{trust}^{update}). During the reevaluation period, if the node behaves well, redemption is awarded, and vice versa. If a node has sufficiently low energy, it may choose to remain selfish to save its energy in a similar way as in transition T_SELFISH. No redemption service is provided for compromised nodes, whether they are detected or not.

5. Numerical Results and Analysis

In this section, we show numerical results obtained by evaluating our hierarchical SPN model. Table 1 gives the default parameter values used in this case study.

Table 1: Default parameter values used.

Parameter	Value	Parameter	Value	Parameter	Value	Parameter	Value
$\boldsymbol{N}$	160	S_{init}^{NT1}	(3, 5] m/s	E_{init}^{NT1}	(12, 24] min.	α	0.8
$\boldsymbol{k_{recom}}$	3	S_{init}^{NT2}	(5, 10] m/s	E_{init}^{NT2}	(24, 36] min.	T_{gc}	60*2 s
$\boldsymbol{R}$	250 m	S_{init}^{NT3}	(10, 15] m/s	E_{init}^{NT3}	(36, 48] min.	T_{IDS}	60*10 s
$\boldsymbol{\lambda}$	1/(60*60)	S_{init}^{NT4}	(15, 30] m/s	E_{init}^{NT4}	(48, 60] min.	T_{trust}^{update}	60*2 s
$\boldsymbol{\mu}$	1/(60*60*4)	λ_{com}	1/(60*60*2)	$P_{fn}^{IDS} = P_{fp}^{IDS}$	0.5%	TC	3

The total number of nodes in the network N is set to 160, with 40 nodes for each of the 4 node types. Speed and initial energy level of nodes are selected from a uniform distribution, thus reflecting the heterogeneous characteristic of military environments. The IDS false positive and false negative probabilities were set to 0.005, assuming that a high quality distributed IDS is deployed in the network. A trust update interval of 10 min. is used to monitor network dynamics and update trust values of nodes in the network. The length of the trust chain (TC) is set to 3 because it has been shown that TC=3 provides the maximum trust values [6]. Here, we focus on comparative analysis of MDTM and UFTM.

The key parameters are the weights for QoS trust (β_1) and social trust ($1 - \beta_1$) that can be adjusted for MDTM and UFTM. UFTM uses identical weights (QoS trust: social trust = 0.5: 0.5) to consider both QoS trust and social trust properties with equal importance regardless of the characteristics of the mission. On the other hand, MDTM uses a different set of the ratio of QoS trust and social trust in order to best identify well qualified nodes based on the characteristics of a given mission. We experiment with two mission scenarios where one scenario (called the QoST mission) requires high QoS trust levels (QoS trust: social trust = 0.8: 0.2 with $\beta_2 = 0.8$) while the other scenario (called the ST mission) requires high social trust levels (QoS trust: social trust = 0.2: 0.8 with $\beta_2 = 0.2$).

Table 2: Trust threshold values used for trust-based reliability calculation.

	QoST mission				ST mission			
	$D_{NT_v}^{j-QoST-1}$	$D_{NT_v}^{j-QoST-2}$	$D_{NT_v}^{j-ST-1}$	$D_{NT_v}^{j-ST-2}$	$D_{NT_v}^{j-QoST-1}$	$D_{NT_v}^{j-QoST-2}$	$D_{NT_v}^{j-ST-1}$	$D_{NT_v}^{j-ST-2}$
$\boldsymbol{NT_1}$	0.6	0.5	0.5	0	0.5	0	0.6	0.5
$\boldsymbol{NT_2}$	0.65	0.5	0.5	0	0.5	0	0.65	0.5
$\boldsymbol{NT_3}$	0.7	0.5	0.5	0	0.5	0	0.7	0.5
$\boldsymbol{NT_4}$	0.75	0.5	0.5	0	0.5	0	0.75	0.5

Table 2 summarizes the trust threshold values used in Equation 9 to calculate the mission success probability *R(t)*.

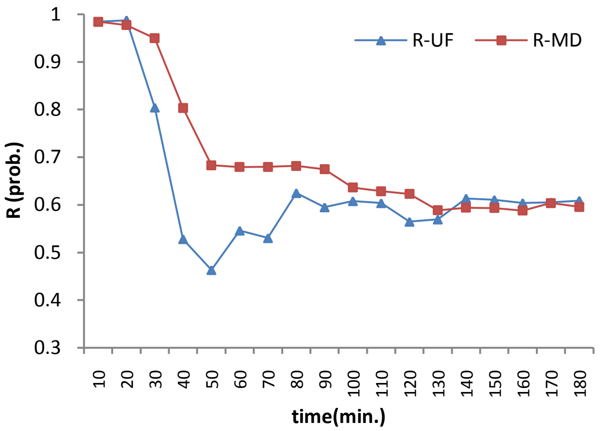

Figure 4: Trust-based Mission Success Probability under QoST mission.

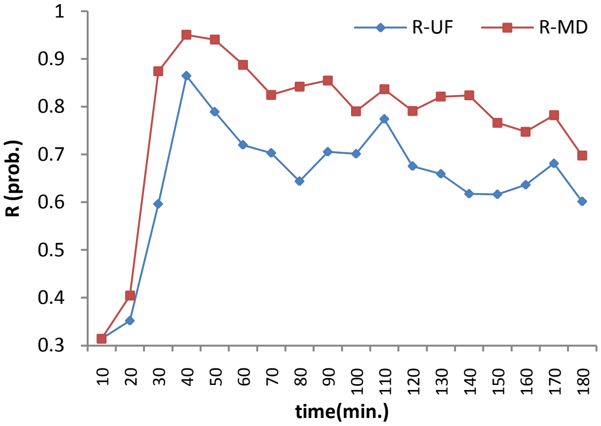

Figure 5: Trust-based Mission Success Probability under ST mission.

We perform a comparative analysis in terms of *R(t)* obtained under MDTM versus UFTM, noted as R-MD and R-UF shortly in Figures 4 and 5. Figure 4 shows *R(t)* over time for the QoST mission. We compare MDTM with $\beta_1 = 0.8$ against UFTM with $\beta_1 = 0.5$, given a QoST mission with $\beta_2 = 0.8$. Figure 4 shows that as time progresses, MDTM performs better overall except when time *t* is sufficiently small (i.e., *t* < 30 min.) or sufficiently large (i.e., *t* > 130 min.). The reason is that for the QoST mission in Figure 4, MDTM will select more QoS qualified nodes that are usually less selfish and consume more energy by cooperating more than those selected by UFTM. As time progresses, the energy, particularly for nodes of node type 1 (type 1 nodes have the least energy), will be drained and they become selfish to save energy. On the other hand, UFTM will select nodes with the same weights for both QoS trust and social trust and accordingly nodes still having high energy may not be selected compared to those selected by MDTM. Thus, in UFTM, energy will not be drained as much as in MDTM and nodes with relatively more energy will remain alive compared to MDTM even as time progresses. Consequently, UFTM could perform better than MDTM at large *t* because it has more nodes left with high energy.

Figure 5 shows *R(t)* over time for the ST mission. We compare MDTM with $\beta_1 = 0.2$ against UFTM with $\beta_1 = 0.5$, given a ST mission with $\beta_2 = 0.2$. Except for the point at time *t* = 10 min. where UFTM and MDTM select the same set of nodes for mission execution due to the assumption that at time *t* = 0 all nodes are ignorant, MDTM clearly outperforms UFTM. This is because the trust values of social trust components (i.e., honesty, proximity, and betweenness) are mainly affected by randomly selected location information as well as a random seed selected as the nature of the dishonest behavior of a node. Even if we associate a node's dishonest behaviors with the remaining energy level of a node, the values of the social trust components are relatively less affected by energy consumption over time, although we also observe that the trust values in both MDTM and UFTM decrease as time progresses due to energy exhaustion.

To understand how MDTM or UFTM affects *R(t)*, we study how each scheme (MDTM or UFTM) dynamically selects a set of qualified nodes during each trust evaluation period. To effectively show the membership dynamics we used the Mean Percentage Difference (MPD) to explain the membership difference between MDTM and UFTM, with a larger MPD indicating a higher dynamic membership change of MDTM over UFTM. The MPD is computed by:

$$MPD^{|MD-UF|}_{members\ hip-NT_v} = \frac{\sum_{i\in S}\frac{\sum_{k\in G}\left|M^{iR-MD}_{k,\,NT_v} - M^{iR-UF}_{k,\,NT_v}\right|}{|G|}}{|S|} \qquad (18)$$

where $M^{iR-MD}_{k,\,NT_v}$ or $M^{iR-UF}_{k,\,NT_v}$ represents the membership status of node *k* in node type *v* in each scheme, with 1 for a selected member, 0 otherwise. $|S|$ is the number of iteration rounds (or the number of trust evaluation periods), $|G|$ is the number of nodes in each node type *v*, and *k* is the node ID in node type *v* at the *i*th evaluation period.

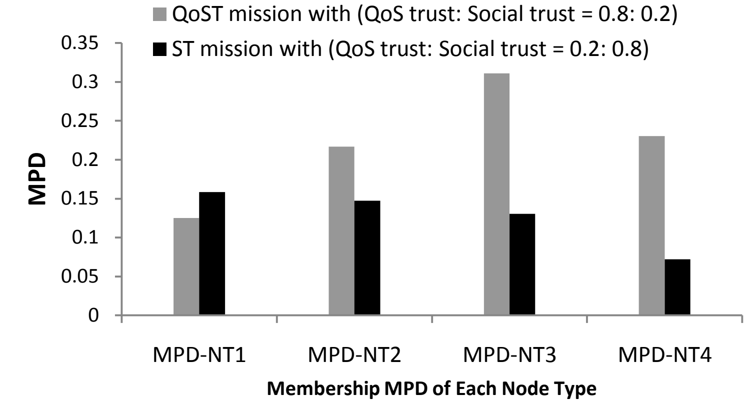

Figure 6: MPD based on the membership dynamics of MDTM and UFTM in each node type under QoST mission and ST mission.

From Figure 6, we notice that the MPD is larger for the QoST mission except for the least-capable nodes (type 1), whereas the MPD is larger for less capable node types with the ST-type mission. Further, the QoST mission shows more dynamic membership changes between trust evaluation periods while the ST mission shows more stable memberships in both schemes. Note that a high MPD indicates a high membership difference between MDTM and UFTM.

6. Conclusion and Future Work

We proposed a composite trust metric comprising *QoS* trust derived from communication networks and *social trust* derived from social and cognitive networks. We then proposed a mission-dependent trust management protocol suitable for military battlefield tactical MANETs based on the context-dependent characteristics of our composite trust metric. We considered dynamic membership change upon every trust update for selecting a mission team that consists of highly qualified nodes to maximize the mission success probability based on the trust values of nodes selected for mission execution. We developed SPN models for performance evaluation of our trust models and protocols. In particular, we developed a continuous-time hierarchical modeling technique that takes into account network dynamics between trust evaluation periods. Our results showed that the proposed mission-dependent trust management protocol outperforms a unified trust management protocol overall in terms of the mission success probability.

Our future research will extend this work by: (1) identifying the optimal weights for our proposed mission dependent trust management protocol, taking into account operation and mission requirements; (2) implementing multiple concurrent missions with multiple mission teams and analyzing their effects on the mission success probability, membership change, and trust values of mission-participating nodes; (3) considering other types of trust properties such as situational awareness or leadership or group solidarity derived from social and cognitive networks; (4) identifying an optimal trust interval balanced with "jitter and flap," since rapid and continuous changes in team composition may hinder mission success; and (5) validating and verifying the proposed trust models and trust propagation protocols through extensive simulation and, where feasible, experiments in practical environments.

Acknowledgement

This project is supported in part by an appointment to the U.S. Army Research Laboratory Postdoctoral Fellowship Program administered by the Oak Ridge Associated Universities through a contract with the U.S. Army Research Laboratory.

References

[1] W. J. Adams, G. C. Hadjichristofi and N. J. Davis, "Calculating a Node's Reputation in a Mobile Ad Hoc Network," *Proc. 24th IEEE Int'l Performance Computing and Communications Conference (IPCCC'05)*, Phoenix, AX, 7-9 Apr. 2005, pp. 303-307.

[2] C. Bertocco and C. Ferrari, "Context-Dependent Reputation Management for Soft Security in Multi Agent Systems," *IEEE/WIC/ACM Int'l Conf. on Web Intelligence and Intelligent Agent Technology (WI-IAT'08)*, 9-12 Dec. 2008, Sydney, NSW, Australia, vol. 3, pp. 77-81.

[3] H. Billhardt, R. Hermoso, S. Ossowski, and R. Centeno, "Trust-based Service Provider Selection in Open Environments," *Proc. 2007 ACM Symposium on Applied Computing*, 11-15 Mar. 2007, Seoul, Korea, pp.1375-1380.

[4] M. Blaze, J. Feigenbaum, and J. Lacy, "Decentralized Trust Management," *Proc. IEEE Symposium on Security and Privacy*, 6-8 May, 1996, pp. 164 – 173.

[5] J.H. Cho and A. Swami, "Towards Trust-based Cognitive Networks: A Survey of Trust Management for Mobile Ad Hoc Networks," *14th Int'l Command and Control Research and Technology Symposium*, Washington D.C. 15-17 June 2009.

[6] J.H. Cho, A. Swami and I.R. Chen, "Modeling and Analysis of Trust Management for Cognitive Mission-driven Group Communication Systems in Mobile Ad Hoc Networks," *2009 IEEE/IFIP Int'l Symposium on Trusted Computing and Communications,* Vancouver, Canada, 29-31 Aug. 2009.

[7] K. S. Cook (editor), *Trust in Society*, vol. 2, Feb. 2003, Russell Sage Foundation Series on Trust, New York.

[8] A. Corradi, R. Montanari, and D. Tibaldi, "Context-Driven Adaptation of Trust Relationships in Pervasive Collaborative Environments," *Proc. 2005 Symposium on Applications and the Internet Workshops (SAINT-W)*, 2005, pp. 178-181.

[9] S. Corson and J. Macker, "Mobile Ad Hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations," RFC 2501, Jan. 1999.

[10] F. C. Gärtner, "Byzantine Failures and Security: Arbitrary is Not (always) Random," Swiss Federal Institute of Technology (EPFL) School of Computer and Communication Sciences, *Technical Report* IC/2003/20, 2003.

[11] J. Golbeck, "Computing with Trust: Definition, Properties, and Algorithms," *Securecomm and Workshop 2006*, 28 Aug. – 1 Sept. 2006, pp. 1-7.

[12] E. Gray, P. O'Connell, C. Jensen, a. Weber, J.-M. Seigneur, and C. Yong, "Towards a Framework for Assessing Trust-based Admission Control in Collaborative Ad Hoc Applications," *Technical Report*, TCD-CS-2002-66, Trinity College Dublin, 2002.

[13] J. Li, R. Li, and J. Kato, "Future Trust Management Framework for Mobile Ad Hoc Networks: Security in Mobile Ad Hoc," *IEEE Communications Magazine,* vol. 46, no. 4, Apr. 2008, pp. 108-114.

[14] H. Luo, J. Kong, P. Zerfos, S. Lu, and L. Zhang, "URSA: Ubiquitous and Robust Access Control for Mobile Ad Hoc Networks," *IEEE/ACM Transactions on Networking (TON),* vol. 12, no. 6, Dec. 2004, pp. 1049-1063.

[15] G. Mainland, D. Parkes, and M. Welsh, "Decentralized, Adaptive Resource Allocation for Sensor Networks," *Proc. 2nd Conf. on Symposium on Networked Systems Design and Implementation*, 2-4 May 2005, Boston, MA, vol. 2, pp. 315-328.

[16] M. Moloney and S. Weber, "A Context-aware Trust-based Security System for Ad Hoc Networks," *Proc. 1st Int'l Conf. on Security and Privacy for Emerging Areas in Communication Networks-Workshop*, 5-9 Sept. 2005, pp. 153-160.

[17] K. Namuduri, “An Active Trust Model based on Zero Knowledge Proofs for Airbone Networks,” *Proc. 5th Annual Workshop on Cyber Security and Information Intelligence Research*, Oak Ridge, Tennessee, 13-15 Apr. 2009.

[18] T. Pham, D. Verma, and G. Pearson, “Intelligence, Surveillance, and Reconnaissance Fusion for Coalition Operations,” *2008 11th Int’l Conf. on Information Fusion,* Cologne, Germanu, 30 June – 3 July 2008, pp. 1-8.

[19] A. Preece, M. Gomez, G. de Mel, W. Vasconcelos, D. Sleeman, S. Colley, G. Pearson, T. Pham, and T. La Porta, “Matching Sensors to Mission Using a Knowledge-Based Approach,” *Proc. Int’l Society for Optical Engineering (SPIE): Defense Transformation and Net-Centric Systems*, 18-20 March 2008, Orlando, FL, vol. 6981, pp. 1-12.

[20] A. Preece, D. Pizzocaro, K. Borowiecki, G. de Mel, M. Gomez, W. Vasconcelos, A. Bar-Noy, M. P. Johnson, T. La Porta, H. Rowaihy, G. Pearson, T. Pham, “Reasoning and Resource Allocation for Sensor-Mission Assignment in a Coalition Context,” *IEEE Military Communications Conf. (MILCOM 2008)*, San Diego, CA, 16-19 Nov. 2008, pp. 1-7.

[21] H. Rowaihy, M. Johnson, A. Bar-Noy, T. Brown, and T. La Porta, “Assigning Sensors to Competing Missions,” *IEEE Global Telecommunications Conf. (GLOBECOM 2008)*, 30 Nov. – 4 Dec. 2008, New Orleans, LO, pp. 1-6.

[22] R.A. Sahner, K.S. Trivedi and A. Puliafito, Performance and Reliability Analysis of Computer Systems: An Example-Based Approach Using the Sharpe Software Package, Springer, 1996.

[23] M. Tavakolifard, S. J. Knapskog, and P. Herrmann, “Trust Transferability among Similar Contexts,” *Proc. 4th ACM Symposium on QoS and Security for Wireless and Mobile Networks*, 27-31 Oct. 2008, Vancouver, British Columbia, Canada, pp. 91-97.

[24] S. Toivonen, G. Lenzini, and I. Uusitalo, “Context-aware Trust Evaluation Functions for Dynamic Reconfigurable Systems,” *Proc. Models of Trust for the Web Workshop (MTW'06)*, 22 May 2006, vol. 22.

[25] M. G. Uddin, M. Zulkernine, and S. I. Ahamed, “CAT: A Context-Aware Trust Model for Open and Dynamic Systems,” *Proc. 2008 ACM Symposium on Applied Computing*, 16-20 Mar. 2008, Fortaleza, Ceara, Brazil, pp. 2024-2029.

[26] H. Wang, Z. Li, B. Yang, and H. Xia, “A Context-Aware Service Matchmaking Method Using Description Logic,” *2nd IEEE Asia-Pacific Service Computing Conf.*, 11-14 Dec. 2007, Tsukuba Science City, Ibaraki Prefecture, Japan, pp. 26-32.